REMARKS

This paper is filed in response to the first office action. A three (3) month extension of time is also submitted.

The drawing objection is noted. Formal drawings are submitted herewith for the Examiner's consideration and approval.

The paragraph beginning on page 9, line 29 and continuing onto the next page through line 10 has been amended solely to improve readability.

Claims 12 and 15-17 were rejected under 35 USC 112, second paragraph, because the phrase "includes the steps of" was deemed to lack antecedent basis. This rejection is traversed. (Moreover, regarding cancelled claim 12, the rejection is moot). The "includes the step(s) of" formulation is conventional and acceptable in a claim preamble, and the usage complained of here was intended to be in a similar vein. Nevertheless, to reduce the number of contested issues claims 15-17 have been amended to remove the objectionable phrase. The claims should now be deemed in compliance with Section 112, second paragraph.

The Examiner will note that the word "output" in claim 9 was changed to "push" to conform to the wording in the following paragraph of the claim. No new matter has been added.

Claims 1-8 are rejected under 35 USC 103(a) as being unpatentable over Jeffords, U.S. Publication No. 20010042139 in view of Janis, U.S. Patent No. 5,263,165.

Claims 9-17 are rejected under 35 USC 103(a) as being unpatentable over Jeffords in view of Janis, further in view of Phillips, U.S. Patent No. 7,058,696.

Respectfully, these rejections should be reconsidered, as Jeffords (the primary reference) does not implement a distributed agreement protocol based on vector exchange as described in this application. To help clarify why this is the case, claims 1 and 9 have been amended to recite the distributed agreement protocol with more clarity. Admittedly, Jeffords does discuss something the inventors there at [0057] refer to as a "resource manager state vector exchange protocol," but this fortuitous overlap in nomenclature is the only commonality; the reference itself describes a method that differs from that claimed here.

In particular, Jeffords describes a method and apparatus for accessing resource objects contained in a distributed memory space in a communications network. A "resource object" is a software object whose data may be important [0223]. In this system, the distributed memory space is divided into a plurality of memory pools, with each pool containing a collection of resource objects [0221]. A plurality of resource manager objects are also provided, with each resource manager object having an associated set of memory pools and a registry of network unique identifiers for the resource objects in those pools. A resource manager provides a registry of resource objects and a mechanism for communicating with other resource managers [0217]. The patent application also describes that the system affords a so-called "relativistic view" of state of a plurality of objects, where each object generates a state vector representing that object's view of its own state and the state of all other objects [0013]. Using a "state vector exchange protocol," each object sends its state vector to other objects, and each object maintains a state matrix of the state vectors [0057].

Figure 4 in Jeffords illustrates a state vector generated by an object A. In this example, object A thinks itself to be in state 1 and think object B is in state n, object C is in state 3, and object D is in state 2 [0180]. Figure 5 is a representative state matrix generated by the combination of state vectors for objects A through D. In this case, each object has provided a state vector (row) to the matrix. The column vector for object A indicates that object A thinks itself to be in state 1, object B think object A is in state 3, object C thinks object A is in state n, and object D thinks object A is in state 3 [0185].

In Jeffords, it is assumed that a distributed application runs with one or more processes at any given time. On start up, each process initializes an internal contact status matrix (CSM) and attempts to contact a set of other processes. Figure 8 illustrates a method for updating state information in this matrix. The method begins by having a particular process (process 1) update its own contact status vector. Next, the process 1 attempts to contact the other systems. If a contact to another system is successful, the process 1 updates its own state vector to provide this indication. The process 1 then sends out a copy of its state vector to the other systems, e.g., processes 2-n (107). Then, process 1 continues to contact other systems (108); when all processes have been attempted to be

contacted, the method ends. Using the discovery process, each process learns about the relative state of all other processes in the network. In particular, and as described at [0057], this "is done through a resource manager state vector exchange protocol, i.e., the use of state matrices filled by the vectors received at each resource manager ... Once a stable (deterministic) network state has been achieved (all active ... processes are reporting the same state information about each other), synchronization of resource pools is initiated."

Putting aside the admitted nomenclature overlap, it should be appreciated that Jeffords "resource manager state vector exchange protocol" is not a distributed agreement using vector exchange [VE], a preferred embodiment of which is described in the specification [see page 12, line 25+, emphasis supplied] as follows:

"VE generates a knowledge vector for the file. At step 312, this vector is then exchanged among the servers to determine whether a consensus can be reached. In a preferred embodiment, the servers fill the bits of the vector as they learn about it. Thus, for example, server A is initialized as follows:

test.txt.1.9999.A.03485 A:1 B:0 C:0

This string means that server A knows about the file, while servers B and C do not. When server B receives the vector, server B fills its bit and passes on the vector to its peers, e.g.:

test.txt.1.9999.A.03485 A:1 B:1 C:0

Once server A receives the vector back (which is indicated by a positive outcome of the test at step 314), it realizes that a quorum (e.g., a majority) of servers, in this case, servers A and B, know about the file, because the bits are filled and server A is also able to accept this submission (as it sent the vector in the first instance). ...

The vector exchange guarantees is that if the mechanism accepts a submission, then a quorum (e.g., a majority) of servers have agreed on it. Thus, there are no false positives."

In Jeffords, and as described in the flowchart of Figure 8, an individual process attempts to contact other processes (that will be required for use by the distributed application), and that process updates its own state vector accordingly before transmitting that updated vector to the other processes. In Jeffords, a process does not receive a state vector passed from some other process which it then modifies - e.g., by writing into the received vector its own state information. Moreover, in Jeffords, running the state vector exchange protocol appears to be for the purpose of ensuring that the active processes know when they are reporting the same state information about each other; this enables the system there to initiate operation of the distributed application that will use these processes.

The protocol is not, as in the subject disclosure, establishing a guarantee that the servers (that have modified the bits in the knowledge bit vector) have reached an agreement that a file submit process can now proceed. Indeed, the distributed application in Jeffords has nothing to do with file submission; rather, Jeffords describes a distributed call management application [0209] or a text-based conferencing application [0210].

Neither Janis nor Philips makes up for the deficiencies in the primary reference. In this first instance, Janis appears to have been cited solely because it (like Jeffords) uses the "resource manager" construct. The method described there concerns providing "user access control" within a distributed data processing system. According to the method, a reference monitor service is established. Access control profiles are stored in that service. Thereafter, selected access control profile information is communicated between the reference monitor service and a resource manager in response to an attempted access of a particular resource object controlled by that resource manager. A resource manager may utilize this communication technique to retrieve, modify, or delete a selected access control profile, as desired. Further, the resource manager may utilize this communication technique to control access to a resource object by utilizing the information contained within the access control profile to determine if the requester is authorized to access the resource object and whether or not the requester has been granted sufficient authority to take selected actions with respect to that resource object. The patent does not address file submissions across a distributed computer network, nor does it disclose or suggest any type of distributed agreement protocol, let alone such a protocol using vector exchange.

Philips describes a multi-user file storage service and system that enables each user of a user group to operate an arbitrary client node at an arbitrary geographic location to communicate with a remote file server node via the Internet. More than one user of the user group is permitted to access the file group at the remote file server node simultaneously, but the integrity of the files is maintained by controlling access so that each access to one of the files at the remote file server is performed, if at all, on a respective portion of that file as most recently updated at the remote file server node. The patent is showing for its description of encoding given information about a file into a temporary identifier, and transmitting that identifier. The patent, however, does not

address file submissions across a distributed computer network, nor does is disclose or suggest any type of distributed agreement protocol, let alone a protocol using vector exchange.

There is no evidence in the record that the prior art elements in Jeffords and Janis (and Philips) can be combined according to known methods to yield predictable results; rather the references simply share a common usage of a "resource manager" construct. Even if these elements are similar, it is not seen how the proposed combination of references adds anything of value to Jeffords; Jeffords still lacks the vector exchange protocol having at least the following limitations now positively recited in the claims:

Claim 1:

"in response to receipt of a submission of a file at a given server, accepting the submission at the given server only if a given subset of the set of distributed servers reach an agreement to the submission, where the agreement is determined using a data exchange protocol that includes sub-steps as follows:

passing a bit vector from a first server to a second server, the bit vector including a first indication that the first server has knowledge of the file;

upon receipt of the bit vector at the second server, having the second server modify the bit vector to include, together with the first indication, a second indication that the second server also has knowledge of the file;

having the second server pass the bit vector, which includes the first and second indications, to one or more other servers in the given subset; and

upon a given state being reached, as indicated by at least the first and second indications in the bit vector, determining that the agreement has been reached;"

Claim 9:

"if the file has been successfully pushed to each of the other servers, having the given server initiate a data exchange protocol to each of the other servers to which the given server has connectivity, where the data exchange protocol includes sub-steps as follows:

passing a knowledge bit vector among the given server and the other servers;

having each server that receives the knowledge bit vector modify the knowledge bit vector to indicate that server's knowledge of the file; and

based on the knowledge bit vector as modified, determining whether a quorum of the servers have reached a given state; and

when the quorum of servers reach the given state, accepting the file for submission."

Further, there is likewise no evidence that the Janis "resource managers" could be substituted into the Jeffords system to produce any predictable result or why anyone of ordinary skill would even want to do so, as these systems have different and unrelated purposes (Jeffords being concerned with a distributed application and Janis being concerned with user access management). Moreover, the recited file submission method is not simply the application of a known technique to a known method ready for improvement to yield predictable results, nor is the described subject matter in any way obvious to try or disclosed/suggested by any possible combination of the references.

Further, any purported combination of the references would still lack at least the above-recited subject matter when considered "as a whole" within the meaning of 35 USC 103(a). Thus, the claims recite patentable subject matter.

For these reasons, a Notice of Allowance is respectfully requested.

Respectfully submitted,

/David H. Judson/

By: _____

David H. Judson, Reg. No. 30,467

ATTORNEYS FOR APPLICANT